# Data Engineering Final Project

## Project Taken

**Pre-Requisites**

1. Create a Google Account
2. Created a Project
3. Gcloud SDK installed on the machine
4. Create a Service Account and store the JSON key to the installation path as ‘gcp.json’
5. Run “gcloud auth activate-service-account --key-file=/path/to/gcp.json

**Pending Tasks**

* Fix Terraform
* Visualization as part of DBT pipeline
* Add Data transformer
* Explore keeping Kaggle and GCP JSON in secret
* Adding to Github
* Read.Me documentation
* Add tests
* CI/CD

## Datasets Taken

<https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce?resource=download>

<https://www.kaggle.com/datasets/olistbr/marketing-funnel-olist>

## Use Cases

* <https://www.kaggle.com/code/anshumoudgil/olist-ecommerce-analytics-quasi-poisson-poly-regs>
  + Customer Journey through Olist
  + Increase in number of sellers per month
  + Number of new Product categories added per month
  + Categories Purchased at high frequency yearly, monthly
  + Success of Orders by region
  + Success of Payment Methods by region
* Seller vs Customer spread across states
* Marketing Impact on
  + Addition of new Sellers
  + Addition of new Customers
  + Addition of new Orders
* Predict the lead time to deliver
* Repeat rate of Customers

## Installation Steps

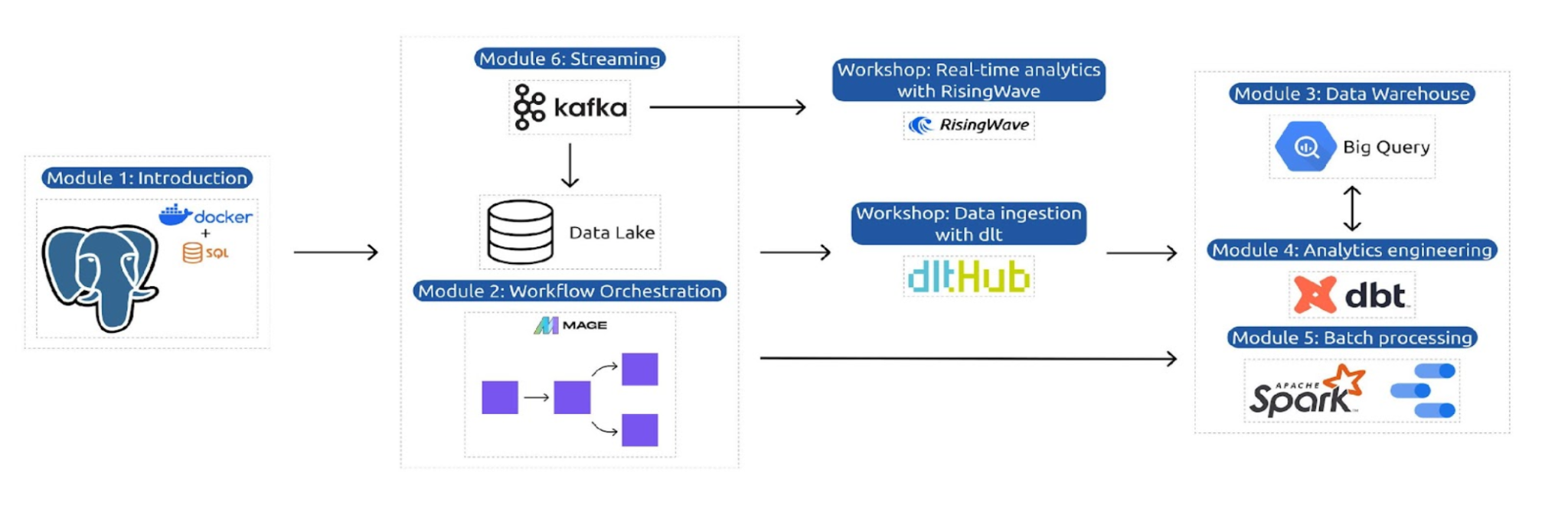
* Create a Google Account
* Create a Project
* Dockerfile
  + Downloaded Dockerfile from <https://docs.mage.ai/integrations/spark-pyspark>
  + Add default-jdk and default-jre and scala and unzip to downloaded dockerfile
* Terraform - In GCP GCP
  + Terraform Setup
    - Copy the Docker file, Installation Script, kaggle.json, bash.rc and requirements.txt
    - Build the Docker image in below steps as mage\_spark
  + docker run -d -t --name mage\_spark -e SPARK\_MASTER\_HOST='local' -p 6789:6789 -v $(pwd):/home/src mage\_spark /app/run\_app.sh mage start skpmagepl
  + docker exec -d -t mage\_spark bash
  + bash installationscript.sh
* Run installationscript.sh to do the following
  + Create folders and directories
  + SPARK installation

<https://www.atlantic.net/dedicated-server-hosting/how-to-install-and-setup-apache-spark-on-debian-10/>

Install Spark

* wget <https://dlcdn.apache.org/spark/spark-3.5.1/spark-3.5.1-bin-hadoop3.tgz>
* tar xzfv spark-3.5.1-bin-hadoop3.tgz
* rm spark-3.5.1-bin-hadoop3.tgz
* mv spark-3.5.1-bin-hadoop3 /opt/spark
* Install Python libraries from requirements.txt
  + - * pyspark==3.5.1
      * pandas==2.2.1
      * dlt
      * dlt[duckdb]
      * dask[dataframe]
      * Kaggle

## Design

****

## Top Issues

* Kaggle initialisation in Mage Data Loader was an issue
  + Conda was installed in Docker.
  + Kaggle and other Python libraries were installed in the Conda Active environment.
  + So when Conda was not installed and python libraries were installed. Data Loader was able to find Kaggle
* Spark Initialization in Mage Data Loader was an Issue
  + DockerFile was created using <https://medium.com/@MarinAgli1/setting-up-a-spark-standalone-cluster-on-docker-in-layman-terms-8cbdc9fdd14b>
  + Spark session had to be initialized using ScratchPad as mentioned in <https://docs.mage.ai/integrations/spark-pyspark#custom-spark-session-at-the-project-level>
* BigQuery tables had ‘Unnamed:0’ as index column.
  + Spark Read CSV to PD.DataFrame added an index column.
  + Referring to <https://sparkbyexamples.com/pandas/pandas-drop-index-column/> and <https://www.kaggle.com/discussions/general/354943>, following were done to remove index
    - pdf = df.toPandas()
    - pdf2 = pdf.reset\_index(drop=True)

## References

**Spark Cluster in Docker**

<https://docs.mage.ai/integrations/spark-pyspark#custom-spark-session-at-the-project-level>

<https://medium.com/@MarinAgli1/setting-up-a-spark-standalone-cluster-on-docker-in-layman-terms-8cbdc9fdd14b>

**Kaggle Datasets**

<https://medium.com/mcd-unison/using-the-kaggle-api-e43e902fba23>

<https://www.kaggle.com/docs/api>

**Terraform remote exec -**  <https://medium.com/google-cloud/terraform-remote-exec-on-google-compute-engine-vm-instance-d47def447072>

<https://github.com/Sayed-Imran/Terraform-Scripts>

<https://github.com/Sayed-Imran/Terraform-Scripts/tree/master/gcp-remote-exec>

<https://gist.github.com/smford22/54aa5e96701430f1bb0ea6e1a502d23a#file-main-tf> <https://www.devopsschool.com/blog/terrafrom-example-code-for-remote-exec-provisioner/>

**Few other Projects considered**

<https://docs.google.com/spreadsheets/d/1YaVYDDlokaaobcjMQvF13kH2mRnTwIbA/edit?usp=drive_link&ouid=112949734852388554027&rtpof=true&sd=true>

**Mage for BigQuery**

<https://github.com/mage-ai/mage-ai/blob/master/mage_integrations/mage_integrations/destinations/bigquery/README.md>

<https://datatalks-club.slack.com/archives/C01FABYF2RG/p1711711901567119>

**Metabase Queries**